Questions
on
QEq
implemented
in Lammps

Daniele Scopece daniele.scopece.science@gmail.com

2014-09-15 started

What is this presentation about?

Here I shortly resume my understanding of the QEq implementation in Lammps.

In addition I write my questions on what I do not understand, so to be hopefully helpful to others.

I refer to the case qeq/slater since I am interested in that.

Quick Resume of Theory

Theory of the Algorithm - 01

The electrostatic energy is given by:

$$E_{es} = E_0 + \sum_{i} q_i \cdot \chi_i + \frac{1}{2} \cdot \sum_{ij} q_i \cdot q_j \cdot V_{ij} \qquad (14.1)$$

The electrostatic interaction is given by:

$$V_{ij} = \int d^3r_i \int d^3r_j \cdot \frac{\rho_i(r_i, q_i) \cdot \rho_j(r_j, q_j)}{r_{ij}}$$
(14.2)

The atomic charge density distribution in a generic point of space is assumed as slater 1s orbital:

$$\rho_i(r, q_i) = Z_i \cdot \delta(r - r_i) + (q_i - Z_i) \cdot f_i(r - r_i)$$
(14.3)

The function f is expressed as:

$$f_i(|r - r_i|) = \frac{\xi_i^3}{\pi} \cdot e^{-2 \cdot \xi_i \cdot |r - r_i|}$$
 (14.4)

Theory of the Algorithm

chizj[i]With these, the terms in eqn. (14.1) become: E_0 = (neglected since included into the non-electrostatic pair) $\chi_i = \chi_i^0 + \sum_j Z_j \cdot ([j|f_i] - [f_i|f_j]) \Longrightarrow (\text{just short range})$ $V_{ij} = J_i^0 \cdot \delta_{ij} + [f_i|f_j]$ $[f_i|f_j] = \left[\frac{erfc}{r} + (...)\right]_{short-range} + \frac{erf}{r}_{long-range}$ (...) = (Expression in zei zej)

Theory of the Algorithm

The minimum of eqn.(14.1) will be reached under the condition of $\sum_i q_i = 0$ when:

$$\sum_{j} V_{ij} \cdot q_{j} = \mu - \chi_{i}$$

$$q_{i} = \sum_{j} V_{ij}^{-1} \cdot (\mu - \chi_{j})$$

$$\mu = \frac{\sum_{ij} V_{ij}^{-1} \cdot \chi_{j}}{\sum_{ij} V_{ij}^{-1}}$$

The system in eqn.(14.11) is solved in two subsystems:

$$\sum_{j} V_{ij} \cdot s = -b_s = -\chi_i$$
 $\sum_{j} V_{ij} \cdot t = -b_t = -1$

They are solved in Lammps one after the other.

s and t are fictitious charges. The final charge is given by:

$$q_i = s_i - \mu \cdot t_i = s_i - \frac{\sum_i s_i}{\sum_i t_i} \cdot t_i$$

Theory of the Algorithm

To make the code faster a pre-conditioning can be done on Vij by considering only the its short-ranged part:

$$V_{ij} = V_{ij-s} + V_{ij-l}$$
 (14.17)

$$V_{ij-s} = J_i^0 \cdot \delta_{ij} \qquad (14.18)$$

$$V_{ij-l} = [f_i|f_j]$$
 (14.19)

$$V_{ij-s}^{-1}$$
 = Hdia_inv[i] = $1/J_i^0$ = 1/eta[itype]

Questions

H in lammps = Vij here

The complete H from theory would be $V_{ij} = J_i^0 \cdot \delta_{ij} + [f_i|f_j]$

$$V_{ij} = J_i^0 \cdot \delta_{ij} + [f_i|f_j]$$

Inclusive of both short and long-ranged

The short-ranged terms of the matrix H is computed in the subroutine calculate H:

```
273
     etmp1 = 1.00 * (ci_jfi - ci_fifj);
     etmp2 < 0.50 ) (ci_fifj + erfcr*rinv);
274
275
      zjtmp += qqtd2e * zj * etmp1;
276
277
      if (DS debug) printf("\n-fix qeq slater.cpp :: -----
278
      return qqrd2e * etmp2;
```

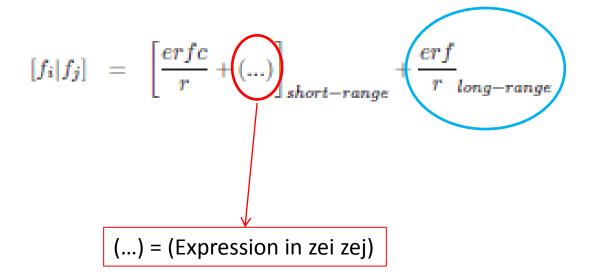
$$H.val[m_{fill}] = kc \cdot \left| \frac{1}{2} \right| \cdot \left([fi|fj]_{lazic} - \frac{1}{r} + \frac{erfc}{r} \right)$$

Why is this factor ½ here? It is not present in the equation ...

These seem just the short-ranged terms of the electrostatic potential.

Where are the long-ranged terms (in reciprocal space?) computed?

I couldn't find them ...



In the function sparse_matvec, the matrix H is multiplied by x and the result stored in q.

The final expression is:

$$q[i] = \left[J_i - 2 \cdot kc \cdot \left(\frac{1}{2} \cdot \frac{erfc}{r}\right) + \left(\frac{1}{\sigma \cdot \sqrt{\pi}}\right)\right] \cdot x[i] + \\ + \sum_{j} kc \left(\frac{1}{2}\right) \left([fi|fj]|_{lazic} - \frac{1}{r} + \frac{erfc}{r}\right) \cdot x[j]$$
 «self» part of the long-range interactions

(This comes from the variable woself)

Where does this term come from?

Why does the erfc appears twice in this expression?

Where is the long-ranged (reciprocal space) term here?

In the file fix_qeq_slater.cpp is also a function called calculate_H_wolf

But in the examples online this is not invoked.

What is that for?
In what case is it invoked?

Thank you for your help